Situated Computing: The Next Frontier for HCI Research

Kevin L. Mills
Information Technology Laboratory
National Institute of Standards and Technology

Jean Scholtz
Information Technology Office
Defense Advanced Research Projects Agency

Abstract. Increasingly, our modern, mobile population works and lives with information. Most individuals interact with information through a single portal: a personal desktop or laptop computer. To provide mobile workers with more convenient access, companies are beginning to produce various portable and embedded information devices. These developments hint at a future where people will interact with information through a continuously varying array of devices that combine to form ad hoc portals suitable to particular situations. In such a future, people and information will be emancipated. No longer will information be captive of single devices, nor will one person necessarily own each device. This leap of imagination requires that human-computer interaction (HCI) researchers solve some significant challenges. This paper identifies and discusses these challenges, and also points to some current, early research on the trail to the next frontier of human-computer interaction.

Introduction

Increasingly people work and live on the move. To support this mobile lifestyle, especially as our work becomes more intensely information-based, companies are producing various portable and embedded information devices. Consider for example, personal digital assistants (PDAs), cellular telephones, pagers, active badges and intelligent buttons. Cellular phones allow us to receive and place telephone calls

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **2002**	2. REPORT TYPE	3. DATES COVERED **00-00-2002 to 00-00-2002**
4. TITLE AND SUBTITLE **Situated Computing: The Next Frontier for HCI Research**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **National Institute of Standards and Technology,Gaithersburg,MD,20899**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
Chapter 24 in Human-Computer Interaction in the New Millennium, (John M. Carroll, editor), ACM Press, Addison-Wesley, New York, 2002

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**25**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

anywhere. Personal Digital Assistants let us take calendar information, contact information, and even e-mail messages with us when we leave the desktop. Active badges and intelligent buttons give us ways to track objects and people. Carrying the idea of a mobile information device toward a natural extension, in 1997 Daimler-Benz announced the demonstration of a concept car: Internet Multimedia on Wheels [18]. In this concept, a car would become a node on the Internet, allowing information services to be delivered to the car and back using wireless technology. Interesting wireless technologies, including Bluetooth [16], IrDA [22] (Infrared Data Association- standards for infrared communications) and HomeRF ™ [21] (wireless home networking), promise to outfit portable and embedded devices with high-bandwidth, localized wireless communication that can also reach the globally wired Internet.

An impressionist painting emerges of nomadic workers with collections of small, specialized devices roaming among islands of wireless connectivity within a global sea of wired networks. Each wireless island defines a context of available services, embedded devices, and task-specific information. As nomadic workers roam the landscape the context in which they are working continuously changes. As workers move onto wireless islands of connectivity, their context is merged with the context of the island to automatically compose a computational environment to support their needs. At other times, when not connected, an array of portable devices provides each nomad with a local context for computing. This painting, which relies heavily on Weiser's [47, 48] concept of ubiquitous computing and on Suchman's notion of situated computing [44], suggests a future where information and people connect directly and work together across a range of contexts.

Weiser envisioned a future where people would interact continually with computation embedded in physical objects. The computers would be small enough to be invisible inside the physical objects and would enhance, rather than interfere with, the original functionality of the physical objects. In Weiser's vision, people would do their work assisted by computer technology, but without having to focus on the computers. This vision continues today in Don Norman's prospect for the invisible computer [33]. Suchman goes further, suggesting that not only should the computer step into the background but also that the computer should continuously monitor the situation in order to proactively aid an information user [44]. Aiming to improve our interaction with information, researchers today investigate four main directions: Smart Spaces or Smart Rooms, Wearable Computing, Tangible User Interfaces, and Information Appliances. While each of these directions shows promise along some dimensions of ubiquitous computing, they fail along others. We will discuss these research efforts later in the paper. First, though, from the shortcomings of this current research, we discern two grand challenges that prevent the universal use of ubiquitous computing.

As a first grand challenge, researchers must alter the inequality of interaction between the two participants: the human and the computer. Currently, the human is responsible both for manipulating and managing the information; that is, locating the information, synchronizing the information, moving the information between devices, and possibly converting the information to a format required by a given device or application. The human is clearly the active player, while the computer assumes a more passive role. This inequality must be altered so that people need only interact with their information, while the computer takes on the ancillary management tasks. As grand

challenge two, researchers must find a means to endow cyberspace with a better understanding of the physical and logical world in which people live and work. Moreover, the computer needs to understand and adapt to the user. In order to accomplish this, researchers must give the computer knowledge of the user's context - the task, the environment, the user's emotional and physical state, and the available computing resources. To be truly invisible, the computer needs to gain an understanding of context without relying on the user to supply that information.

In this paper, we outline specific facets of these two grand challenges. We assert that the human-computer interaction (HCI) research community must meet these challenges before society can reap full benefits from specialized, information appliances. In the sections that follow, we discuss some specific research problems that must be solved to meet each grand challenge. Where applicable, we also point to some ongoing research that appears to be tackling, at an early stage, some aspects of these challenges.

Grand Challenge #1: Emancipating Information

Today people collect information in spreadsheets, databases, document repositories, and web sites. In the main, each set of information is captive of a specific application program. The application dictates the format of the information, and provides the means of interacting with the information. To move data between computers in an understandable form, industry has agreed to a uniform approach, based on Multipurpose Internet Mail Extension (MIME) types, which permit an electronic mail message to describe the format of any included attachments. Even when a computer understands the type of specific attachments, appropriate software must exist on the receiving node in order for the data to be useful. For example, to move data between different types of applications (such as spreadsheet to document) or between different products for the

same type of application (such as Microsoft Word™ to Lotus Wordpro™), either the information must be exported and imported through compatible filters, or the information must be encapsulated inside information of another type, but in a form (such as Microsoft Object Linking and Embedding) that enables the appropriate application to be initiated when a user selects the encapsulated information. In addition to application programs controlling information, the applications themselves are captive within specific computer operating systems. For example, while Microsoft Word will certainly execute on Windows 98™ or Windows NT™, the application will probably not execute on Sun Solaris™. These captivating dependencies will become even more irksome as people begin to use the myriad of specialized devices, such as cell phones, personal digital assistants, pens, pads, and wristwatches, to collect, view, and transport information. The need for information filters and data synchronization programs will increase rapidly. As a result, if the current paradigm continues, then people will be spending more unproductive time managing information, that is, locating data, transforming it to an appropriate format, and sending it to an appropriate device.

In the past, industry has developed standards for describing data for various applications, such as the office document architecture (ODA) [23] and office document interchange format (ODIF) [19] for professional documents. For some reason, these past attempts at uniform data-description languages have failed in the market place. Industry continues to explore alternative technologies, such as eXtensible Markup Language (XML™), which can provide more precise information about the structure and format of data. Successful development of XML as a universal data-description language might one day enable every application to provide a single import and export filter; thus, removing

the current cacophony of filters deployed with each application. Even in the case of XML, competing approaches are emerging for encoding information intended for exchange over wireless communication channels, as distinct from wired Internet channels. Further, assuming that XML is universally deployed to describe data, various applications must still act on the data in order to provide behavior. No widely accepted approach exists to describe behavior appropriate to specific data. Java™ [25] and other platform-independent languages, such as Python [38] and TCL [34], show one possible approach to solve the problem of expressing behavior. An alternate possibility envisions treating software behaviors more as a network service. In such cases, once an appropriate description of the data exists, behaviors can be located as services on the network. For example, Microsoft recently unveiled their vision of a next-generation Windows service architecture. Success in such endeavors will require widespread, almost universal, agreement on the techniques for expressing data format. Perhaps XML will achieve this goal. The second requirement for success entails a means to associate behavior with data. One approach requires all nodes and devices to include a run-time environment that can interpret behaviors described in a standard language. Another approach requires data to include references to behaviors that can be located on the network. In the past, these objectives have proven difficult to achieve, though some progress can be discerned.

To understand the extent of the problem better, consider the study that Jun Rekimoto made of software engineers, arguably among the most advanced users of computer software [39]. Among the software engineers surveyed, Rekimoto found that 54% had three or more computers on their desks, 39% had two computers, while the remainder had only one. Seventy percent of those engineers transferred data between

computers very often and another 25% transferred data often. When considering only nearby computers, 28% of the engineers moved data very often, 23% often, and 36% sometimes. Transfer mechanisms included cut-and-paste, shared files, file transfer, e-mail, and floppies. The decisions about what information to transfer and where, and the means of transfer were all left to the software engineers. While this data comes from a highly specialized user community, we expect that many users, less skilled than these software engineers, will soon face such problems on a daily basis, concomitant with the increase in specialized information devices.

Aside from the overhead of managing our increasingly scattered information, we are all becoming more mobile in our working lives. For example, Bellotti and Bly studied the work activities of a product design team in a company with various facilities distributed around a small geographic area [4]. In particular, the study identified the places where designers did their work, and measured how much time they spent in each place. For two typical product design engineers, Bellotti and Bly discovered that only 10%-13% of the designer's work was conducted at their desktop computers, while 76%-82% of the work was spread over 11 other locations, and 8%-11% of work time was spent moving between work locations.

For our purposes, two observations are worth nothing from the Belotti and Bly study. First, as workers move among work locations they must carry with them a range of information and portable tools that will be needed at each work site. Second, at each work site, there exists a number of local tools, and perhaps some relevant local information, as well as tools and information brought by others on the design team. The designers must combine the local tools and information with the imported tools and

information in order to complete specific design tasks. While these designers probably represent an extreme focus on mobility, we argue that an increasing population of workers spends more time at different locations and traveling among locations. Even within a more typical office environment, workers attend meetings in conference rooms, visit colleagues in their offices, and discuss work over lunch in the cafeteria.

We see new work styles emerging where people will increasingly: (1) move among locations to complete work, (2) use a number of specialized, portable and embedded devices in ad hoc arrangements at each work location, and (3) shuffle information back and forth among work locations and among devices. For such work styles to prove productive, the information technology research community must liberate information from the confines of specific applications and specific computers. We discuss in the following paragraphs some ideas necessary to support these new work styles.

Moving Information to People. One option is to carry all of our information with us. This approach appears feasible, as the miracle of hardware continues to bring us ever-increasing density in disk storage, along with cheaper and faster processors. We don't believe, however, that this will prove feasible because human activities continue to produce information at prodigious rates, and not all such information belongs to particular individuals. In fact, much of the information we produce is context-dependent. For example, we typically attend meetings to conduct specific tasks. Before, after, and during these meetings we create information. Some of this information we retain personally, while other information is shared among the meeting attendees and others outside of the group. Only a small fraction of this information is our own personal

information. Surely, as we move to the next meeting on the same subject we will wish to have information from the last meeting available.

We argue that context can often be inferred from a combination of user, location, and task. If so, then why should a user be required to ensure that the right information is available at the right place and time? Can't the information itself take on this responsibility? Imagine ***active information*** objects that can move, that can replicate themselves, and that can communicate as a group. Active information objects should monitor context and remind us of their existence. Wouldn't it be useful to have your information remind your workgroup that you had discussed the same topic several weeks ago and present you with a summary of that discussion? Active information should be able to track the location, state, and trajectory of information users, of object replicas, and of linked objects. In addition, active information objects should be able to plan the movement, replication, and transformation of information to serve the projected needs of its users. Active information objects must also be able to implement consistency, access, and sharing policies among replicated and linked objects.

A combination of commercial and research activities show some promise that a day will soon appear in which active information becomes both possible and interesting. Clearly mobile code systems, such as Python and Java, hint at the possibility of distributed object systems that can replicate and move. The computer science research laboratory at UC Berkeley [5, 31] is developing scalable reliable multicast protocols, beaconing protocols, and transcoding algorithms that distributed objects can use to discover each other, to communicate, and to transform their presentation. Other work at UC Berkeley promises a processing-capable network infrastructure that can provide a

platform for mobile distributed objects to reside within a network and to move and copy themselves toward specific situated computing locales as users begin to congregate [2]. The OceanStores [5] work on persistent storage, also at UC Berkeley, aims to define secure, reliable storage for a ubiquitous computing environment. By using unique identifiers for the data, encrypting the data, and providing multiple paths to locate data objects, a nomadic worker would be able to access data from anywhere, assuming Internet connectivity.

Novel research is still required to investigate information models that will make it possible for information to transform itself for specific contexts, including the applications available, the devices and other resources at hand, and the tasks to be performed. In addition, information objects will need mechanisms to reveal their active properties and to discover the active properties of other information objects in order to permit individual objects or object webs to combine into larger object systems to support specific contexts and tasks. One particular active property must describe the mechanisms through which users can interact with specific information objects, independent of particular devices and applications.

Removing the Tyranny of an Interface per Application per Device. As many specialized devices become available, human-information interfaces can be distributed across devices and interaction modes. In fact, several devices can be networked to support a richer interaction and computing capability than any of the single devices alone. We use the term **multi-modal** to refer to interfaces that combine modes of interaction. In today's user interfaces, multi-modal most often refers to two modes of input, typically pen and speech. To make interactions in a ubiquitous computing environment truly

natural, this capability must be extended to include gestures, facial expressions, gaze, and tactile input, among others. Multi-modal should include a combination of multiple modes of interaction, where multiple is greater than two!

Depending upon application requirements, user preferences, and knowledge about human awareness, about specific tasks, and about the type of information being conveyed, tomorrow's multi-modal interfaces must coordinate interactions across devices and among interaction events. In addition, a model of interaction events will be needed, as well as rules for mapping between the interaction event model and mode-specific interactions. Given a fluid set of devices available in any particular situated computing locale, software mechanisms must support the dynamic composition of interfaces from among software components and information objects. In addition to composition, rules must also be provided for instantiating optimal multi-modal interfaces for specific tasks, given available devices and modalities. Naming and identification will be a key issue, along with authentication and access control. Since information and interaction events will likely fly through the air across wireless links, privacy will also become more important. Other issues will arise regarding arbitration of shared access to devices within a situated computing locale.

All of these changes have ramifications for the future of software architectures. First, future software architectures for flexible multi-modal interfaces must be constructed from components that will need to discover in real-time a distributed component bus within each specific locality and to configure themselves into the bus. Second, components must be able to discover related components, as well as their capabilities, and to participate in a composition of components into larger services. In

many cases, the capabilities must express assumptions and goals regarding performance, and composition techniques must consider the overall performance requirements of the flexible interface when connecting components together. Third, client components must be prepared to operate robustly in the face of missing or sub-optimal service components. Fourth, components must expect to interact through loosely coupled communication mechanisms that can exhibit various error properties. Industry is developing several competing technologies (e.g., Jini [26] and Universal Plug-and-Play [50]) that could serve as a basis on which to construct tomorrow's flexible, component-based interfaces. HCI researchers should investigate how these technologies can be exploited, extended, and improved to provide the capabilities needed to build effective multi-modal interfaces.

Some researchers are already looking into a few of these concerns. Multi-modal interaction is going beyond speech and pen based interaction. For example, the Rutgers CAIP (Computer Aids for Industrial Productivity) Center has integrated into a single desktop interface a range of multi-modal technologies, including gaze and gesture tracking, voice recognition and speech synthesis, along with the typical display, mouse and keyboard [32]. Visual tracking is also being investigated as an interaction technique [49]. Gestures and facial expressions may soon be used as interaction mechanisms. Wouldn't a confirming nod of the head be even easier at times than saying "yes?" Novel research is still needed to develop an abstract interaction event model that exists independently from specific HCI hardware. In addition, mappings must be developed between the abstract model and specific HCI hardware, both current commercial hardware and experimental hardware. XML might become a specification language that can be translated to appear on different output devices. XML tags and attributes can be

attached to text and then translated at the time that text is to be displayed. Transducers and layout engines are being used to translate web pages so that users of handheld devices can obtain web data. More research is needed into the specification of interactions, independent of device and modality. Such specifications would allow one interface to be developed for use with any input modality and any type of output display.

Information interaction: making it real again. Today we interact with digital information through graphical user interfaces (GUIs) in the WIMP (Windows, Icons, Menus, and Pointers) style. In other words, we use abstract symbols to represent information objects and we manipulate those abstractions. Meanwhile, people have a long history of using physical information objects - books, photographs, newspapers, unstructured notes, and video recordings to name a few. People manipulate these physical objects separately, and then need to execute intermediary translators (such as optical scanners) to bring this “real world” information into the virtual world. Some interesting research seeks to bridge the gap between the real and virtual worlds. Fitzmaurice [10] investigated using objects in the physical world as anchors for digital information. Using handheld portals, people could move through the environment, viewing digital information based on the spatial characteristics of their handheld. Moving the handheld closer to the physical object might cause a computer to zoom in on the information. In a hands-free approach, Steven Feiner and colleagues [51] at Columbia attempt to exploit augmented reality interfaces as a means of relating virtual information with the physical world. At the MIT Media Laboratory, Ishii [27] has been researching tangible user interfaces (TUIs), where physical objects are used to manipulate electronic information with the goal of reducing the cognitive overhead associated with using

electronic information. To the extent that tangible user interfaces build on current user expectations of manipulating physical information, TUIs show promise.

Other researchers also investigate the gap between the real and the virtual. For example, Harrison, et al, [13] at Xerox PARC have investigated user interfaces that exploit physical manipulations to control devices, such as PDAs. Rather than using an artificial input device, such as a mouse or track point, manipulation of the device itself is used as a control. Harrison, et al, have also investigated electronic staples, bits of electronic information embedded into physical objects [14]. They have illustrated this technique using books and posters that advertise events. Here, using a reader attached to a portable computer, the information contained in the staple (a URL in the PARC examples) can be captured by mobile users. Arai, et al, [3] also developed technology that allows people to insert electronic links into paper documents. Len, et al, [29] are developing an electronic environment that allows an interface designer to sketch a user interface, a job that is usually done on paper. In the Portolano project, [24] researchers at the University of Washington instrument biology lab equipment to capture fine-grained experiment details directly from the skills performed as researcher conducts experiments.

Several products being sold today also address the merger of real and virtual worlds. One example is the Cross Pad,™ which combines regular paper and a digital pen with automated capture of digital information. As a user writes on the paper, electronic signals of the strokes made with the pen are stored digitally. After the user returns to a desktop computer, the digital version of the notes can be transferred, as bitmaps, to the desktop. Optical character recognition routines can translate the bitmaps into editable documents.

Grand Challenge #2: Clueing in those Clueless Computers

Norman [33] advocates the use of information appliances, special-purpose computers designed for a particular use. The key principle of an information appliance is simplicity. As the appliance is designed to do one task, that task can be carried out extremely easily. The user does not have to look through many menu options or to supply a variety of information via dialogue boxes. This approach provides a definite improvement over the complexity of our current desktop computer. However, complexity has not vanished - it has merely been pushed to another level. If we want to do more than one task (as most of us must), we now have to decide which appliance to use for which task. Moreover, information for various tasks has to be located and, in some instances, transferred between devices. For example, my pager, my cell phone, my e-mail, and my personal organizer don't know of the existence of each other. If I get an urgent e-mail and don't attend to it within a specified time wouldn't it be useful if I got paged, or if my cell phone called me and read me the message? When I look up a contact on my personal organizer, shouldn't the phone number automatically move into my cell phone? One solution might be to simply combine devices. But where would this stop? Might we wind up with numerous devices hardwired together - yielding a device now as complex as our desktop computer?

How can this complexity be addressed? During any given period of time, a nomadic worker knows that tasks must be performed. Given the worker's preference and the appropriateness of devices for the tasks to be performed, could an appropriate set of devices be assembled by the worker? Perhaps these devices could be interconnected using wireless technologies. This solution would give the worker a custom "wearable"

network of devices. As the selected devices discover each other, they could become aware of the services and information each can provide, and they could combine to support the user's information needs.

At present, such networked-based computing works largely because people carry in their heads a reasonably good model of cyberspace. We know where computers and printers can be found; we know how information can be organized for storage on a disk; we understand the meaning of the three character extensions that tag each file name. We know, but just barely, how to locate, download, configure, and execute various plug-ins to display information in specific formats or to convert information between formats. In fact, sometimes we think our computers and networks should pay us because we sure do a lot of work for them. We also have a good understanding of our environment and how to act in it. Most of the time we remember to shut off our cell phones before the movie starts. We turn down the sound on our laptops when we start them up in meeting rooms (don't we?). We sort out the messages on our pagers, we respond to the urgent, and we defer the less urgent. This sorting represents management overhead time – why should we spend so much time dealing with such mundane tasks? Suppose, on the other hand, that our computers and networks had a much better model of the world in which we, and they, live. Would it be possible for our computers and networks to help us more than we help them today?

To achieve such a world, computer software must begin to understand the context of each situated computing locale in which it operates. By context, we mean the connectivity, bandwidth, and services available in a locale, the location of users, devices, and information relative to a locale, and the physical and logical surroundings within and

near a locale, as well as the tasks being performed and the environment in which those tasks are performed. If computer software can ascertain contextual information, then programs and active information can adapt to the situation, especially as the situation changes when network resources and services come and go and when people enter and leave a locale. What adaptations might be possible?

Multi-modal interfaces could be designed to accommodate a level of uncertainty about the availability of network connectivity and bandwidth, and about the availability of specific interaction devices. Active information might be designed to present different information or to present information in different forms, depending on the number of users, the available devices and network bandwidth, and the user task. Active information might also move or replicate itself to situated computing locales toward which its user or users are moving. Such movement or replication can ensure that task-specific information becomes available when and where needed with little cognitive investment by the user. In addition, since the information will be proximate to the user's interface devices, interaction latency can be reduced. A user interface could also be designed to modify its behavior, and an active information object could be designed to present itself differently, depending upon sensory information about the user's surroundings and environment.

Adapting Information Delivery Using Knowledge of People, Places, and Devices. We suggest that researchers consider trying to build models that cross the gap between physical and logical space, as we perceive it, and cyberspace, as it exists in our computers and networks. Physical space would include models of the practical geometric limits that humans face in physical spaces. Logical space would include models capturing the way in which we think about concepts. Then models that unify the models of

cyberspace, physical space, and logical space are needed to allow computer programs to reason across these spaces. Suppose we could couple sensor data with resource and scene description languages to model within our computers the physical and logical space that people perceive and understand. If we could, then our software might be able to exploit location, proximity, and visibility of both physical and cyber resources to determine where to deliver specific services for us. In addition, our software might be able to adapt information presentation to the specific characteristics of available devices and services. In fact, more generally, if the software has a model of space that appears reasonably consistent with our own, then we might be able to encode, inside a computer, heuristics similar to those that we now use when reasoning on our own about cyberspace.

Think a bit more about this idea. Sensors of all kinds are becoming cheaper and more capable. These include digital still and video cameras, digital sensors, eye-tracking devices, radio-frequency tags, and global positioning system chips. These sensors can be used to determine something about a user's context and to adjust information and its' delivery accordingly. Some context-aware applications already exist. The global positioning systems for automobiles show the position of your car on a map. Georgia Tech’s CyberGuide [1] uses the location of the user, and other tour sites the user has visited, to present appropriate information about the tourist site the user is currently viewing. Other Georgia Tech context-aware applications have used the identity and activity of users to modify the behavior of applications [40]. Using hidden Markov models, researchers at the MIT Media Lab have been able to extract context from ambient audio [8].

Researchers at the MIT Media lab have also tried exploiting location to determine a user's information needs. ComMotion [30] tracks a user via a GPS system and, after having the user identify commonly frequented locations, uses agent technology to monitor incoming messages and queries, delivering the preferred information given the location. Sawhney and Schmandt [41, 42] use the audio level of a user's environment to determine a social context and to deliver e-mail messages in an appropriate mode given the social situation. Researchers have also used context as input. For example, a study of an application built for ecology observations showed that adding contextual information automatically created better data with much less manpower [35].

In addition to simple context information, such as location, researchers are investigating how to account for a user's status when selecting interactions. For example, Picard [37] and Kelin [28] are two researchers investigating concepts of affective computing. In affective computing, the emotional state of a user is considered when determining the computer's formulation of a response. The current work uses various sensory inputs to assess the state of the user. Work on user interface adaptation techniques remains to be explored. A project at Microsoft Research [15] learns about a user's preferences and adapts behaviors accordingly. Still, in the large, much research remains before there exist guidelines regarding interface adaptation given the various emotional states of a user. Progress on these issues may prove crucial for expanded use of wearable computers.

Wearable computers [43, 45] are just now emerging for use by workers carrying out physical tasks, such as aircraft maintenance and inspection and repairing of oil drilling equipment. Today, wearable computers are designed for a specific task or set of

tasks. The form factor for the wearable, the interaction devices, and the information stored on the wearable are all part of the design. If the tasks or the users or the environment in which the task is conducted change significantly, the situation must be analyzed, and a redesign might be required. A redesign can prove costly. If, however, the context of the user could be determined and the interactions modified by the system itself, wearable computing might become less costly. In addition, the design of wearable computers might allow for easy customization by individual users. On first order, a user might assemble a computational device, an assortment of preferred input and output devices, an appropriate set of context-sensing devices, a connection to a wireless LAN, and set off to do the job. Just possibly, user customization might improve the productivity of individual workers.

Solving Three Hard Problems. Enabling customization and adaptation among software elements requires that HCI researchers solve three hard problems. *First, what constitutes context?* What aspects of the task, environment, user, and services need to be considered, and in what detail for what duration? Further, do collaborative tasks have a "collaborative" context, or does everyone have a personal view of the context, or must individual and group contexts by mixed together? As sensors become cheaper, we can begin to capture biological data to augment other context information. The issue is not capturing more data, but identifying the relevant data to capture for various situations.

Secondly, how can context be modeled, represented, and reasoned about in a computer interpretable form? Such models might require multiple levels, and might also entail associating confidence or uncertainty values with interpretations used to construct specific instantiations of the models. Issues of interest might include who is present, what

they are working on, what devices, services, and information resources are nearby, and how these items relate to one another both logically and physically. On a more detailed level, the characteristics and interfaces provided by the available services and devices might also be of use. Researchers will need to devise mechanisms to extract context in both real-time and non-real-time from streams of sensor data, including the ability to derive context when interpreting data across multiple sensor streams.

Once researchers can construct machine-readable models of context, the third tough problem must be faced: *how can models of context be exploited to help users get their tasks completed effectively?* Researchers will need to investigate heuristics for deciding what information to present to users, what devices to use, and what presentation form to select. Of specific concern will be the ability to support dynamic generation of multi-modal interfaces for particular collections of users working in given locations on assigned tasks under varying environmental conditions.

Conclusions

Smart Homes, Smart Cars, Smart Rooms [36] and research projects, such as i-LAND [46], Oxygen [9], Endeavour [20], Portolano [24], and Aura, [17] promise to lead us toward universal ubiquitous computing. Researchers in these, and other, efforts are attempting to devise techniques for computing systems to recognize user activities and to adapt to user needs. While the promise is large, we have a long way to come. We need more concentrated work on context-dependent, or situated, computing. We need projects building large systems, and we also need smaller projects that focus on recognizing and using context, and that look for better ways to bridge the physical-virtual gaps we must currently work around to interact with information. As we build test beds and start living

and working in them, we will solve some research issues but we will also discover others. The initial set of investigations addresses mainly technology issues. While these issues are difficult, the social implications of ubiquitous computing remain largely unknown. What are the implications for our privacy? How will the nature of our work and play change? What are the implications for interaction within and among organizations? Will ubiquitous computing widen or reduce the digital divide?

Today, much of our information-intensive work is carried out at desktop computer workstations. In these settings, the computer is the job. The computing infrastructure supporting our work, including operating systems, applications, and hardware remains relatively stable, and our work location appears mainly fixed. When we adjust our physical environment or when we introduce new applications software or upgrade the operating system or add new networking connections, we expect to spend some amount of our time coping with these changes. Very soon, if not now, we will carry many small information-processing appliances along from place to place as adjuncts to support our jobs. Under such conditions, context changes continuously. If forced to cope with continuous change using the same approach now required for our desktop workstations, we will find that managing our information appliances will become the job. Should that occur, we would cast aside our information appliances. The challenge to the HCI research community is simply this: portable devices and pico-cellular wireless networks are coming in large numbers and quickly, can you provide the foundation needed to extract their value? In the United States and overseas, research funding is now being directed toward various aspects of ubiquitous computing. Within the next five years, large-scale research prototypes will become available for experimentation. For ubiquitous computing

to succeed commercially, these research prototypes must demonstrate an information rich environment with few visible computers.

References

1. Abowd, G. D, Atkeson, C.G., Hong, J., Long, S., Kooper, R. and Pinkerton, M. Cyberguide: A Mobile Context -Aware Tour Guide. *ACM Wireless Networks (3),* 421-433.

2. Amir, E., McCanne, S., and Katz, R. An Active Service Framework and its Application to Real-time Multimedia Transcoding. *Proceedings of ACM SIGCOMM '98*, Vancouver, British Columbia, September, 1998,178-189.

3. Arai, T., Aust, D., and Hudson, S. PaperLink: A Technique for Hyperlinking from Real Paper to Electronic Content. *Proceedings of CHI'97,* Atlanta, GA., March 22-27, 327-333.

4. Belotti, V. and Bly, S. *Walking Away from the Desktop Computer: Distributed Collaboration and Mobility in a Product Design Team,* in *Proceeding of CSCW'96. (*Cambridge, MA) ACM Press, 209-218. 1996.

5. Bindel, D., Chen, Y., Eaton, P., Geels, D., Gummadi, R., Rhea, S., Weatherspoon, H., Weimer, W., Wells, C., Zhao, B., and Kubiatowicz, J., OceanStore: An Architecture for Global-Scale Persistent Storage, to appear in ASPLOS'00.

6. Birbeck, M. *Professional XML*, Wrox Press, Inc., 2000

7. Chawathe, Y., Fink, S., McCanne,, S, and Brewer, E. , *An Active Service for Reliable Multicast in Heterogeneous Environments*. *Proceedings of ACM Multimedia '98*, Bristol, UK, Sept. 1998, 161-169.

8. Clarkson, B., and Pentland, A. Extracting Context from Environmental Audio*, The Second International Symposium on Wearable Computers*, Pittsburgh, PA, October 1998, 154-155.

9. Dertouzos, M. The Future of Computing, *Scientific American*, vol. 281 (2), August 1999, 52-56.

10. Fitzmaurice, G. Situated Information Spaces and Spatially Aware Palmtop Computers. *Communications of the ACM*, 36 (7), July 1993. 38-49.

11. Gessner, R. Netscape's Gecko, *Web Techniques*, March, 1999, 63-70.

12. Haarsten, J., Allen, W, Inouye, J., Joeressen, O., and Naghshineh, M., Bluetooth: Vision, Goals, and Architecture. Mobile Computing and Communications Reviews, Vol. 2 (4), pp. 38-45, 1998.

13. Harrison, B., Fishkin, K., Gujar, A., Mochon, C., and Want, R. Squeeze Me, Hold Me, Tilt Me: An Exploration of Manipulative User Interfaces. *Proceedings of CHI '98*, Los Angeles, Ca., April 1998. pp. 17-24.

14. Harrison, B., Fishkin, K., Gujar, A., Portnov, D., and Want, R. Bridging Physical and Virtual Worlds with Tagged Documents, Objects and Locations. *Proceedings of CHI '99,* Extended Abstracts. Pittsburgh, May 1999. 29-30.

15. Horvitz, E., Breese, J., Heckerman, D., Hovel, D., and Rommelse, K., *The Lumiere Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users.* Proceedings of the Fourtheenth Conference on Uncertainty in Artificial Intelligence, July 1998.

16. Bluetooth SIG web site: http://www.bluetooth.com/default.asp.

17. Project Aura web site: http://www.cs.cmu.edu/~aura

18. Daimler-Benz Internet on Wheels web site: http://www.daimler-benz.com/ind_gfnav_e.html?/research/text/70430_e.htm

19. An ODIF web site: http://www.dsg.harvard.edu/public/intermed/odif-report.html

20. Endeavour Project web site: http://endeavour.cs.berkeley.edu/

21. HomeRF Consortium web site: http://www.homerf.org/tech/

22. Infrared Data Association web site: http://www.irda.org/

23. ITU ODA Specification web site: http://www.itu.int/itudoc/itu-t/rec/t/t411.html

24. Portolano web site: http://portolano.cs.washington.edu

25. Java web site: http://java.sun.com/

26. Jini web site: http://developer.java.sun.com/developer/products/jini/

27. Ishii, H. and Ullmer, B. Tangible bits: towards seamless interfaces between people, bits and atoms. *Proceedings of CHI '97,* Atlanta, 1997. 234-241.

28. Kelin, J., Moon, Y., and Picard, R., This Computer Responds to User Frustration, *Proceedings of CHI '99, Pittsburgh, PA*, May 15-20, 1999. ACM Press, Extended Abstracts, 242-243.

29. Len, J., Newman, M., Hong, J., and Landay, U. DENIM: Finding a Tighter Fit between Tools and Practice for Web Site Design. Proceedings of CHI 2000, The Hague, April 1-6. 510-517.

30. Marmasse, N., comMotion: a context-aware communication system. Proceedings of CHI '99, Pittsburgh, PA, May 15-20, 1999. ACM Press, Extended Abstracts, 320-321.

31. McCanne, S, Brewer, E., Katz, R, Rowe, L., Amir, E., Chawathe, Y., Coopersmith, A., Mayer-Patel, K., Raman, S., Schuette, A., Simpson, D, Swan, A., Tung, T., Wu, D., and Smith, B. , Towards a Common Infrastructure for Multimedia-Networking Middleware, , in Proc. 7th Intl Worskhop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'97), St. Louis, Missouri, May 1997.

32. A. Medl, I. Marsic, M. Andre, C. Kulikowski and J.L. Flanagan, Multimodal Man-Machine Interface for Mission Planning, *Proceedings of the AAAI Spring Symposium on Intelligent Environments*, Stanford, CA, pages 41-47, March 1998.

33. Norman, D., *The Invisible Computer*, Cambridge, MA, MIT Press, 1998.

34. Ousterhout, J. *Tcl and the Tk Toolkit*, Addison-Wesley Pub. Co, 1994.

35. Pascoe, J. Adding Generic Contextual Capabilities to Wearable Computers, *The Second International Symposium on Wearable Computers*, Pittsburgh, PA, October 1998, 92-99.

36. Pentland, A., Smart Rooms, *Scientific American,* vol, 274 (4), April, 1996, 68-76.

37. Picard, R., *Affrective Computing,* Cambridge, MA, MIT Press, 1997.

38. Python Language Web site: http://www.python.org/

39. Rekimoto, Jun. *Pick –and-Drop: A Direct Manipulation Technique for Multiple Computer Environements*, in *Proceedings of the ACM Symposium on User Interface Software and Technology "97*, (Alberta, Canada) ACM Press, 31-39, 1997

40. Salber, D., Dey, A., and Abowd, G., The Context Toolkit: Aiding the Development of Context - Enabled- Applications, *Proceeding of CHI'99*, Pittsburgh, PA, May 15-20, 1999, ACM Press.

41. Sawhney, N. and Schmandt, C. Speaking and Listening on the Run: Design for Wearable Audio Computing, *Proceedings of the International Symposium on Wearable Computing*, October 19-20 Pittsburgh, Pa. 1998

42. Sawhney, N. and Schmandt, C. Nomadic Radio: Scalable and Contextual Notification for Wearable Audio Messaging, *Proceeding of CHI'99*, Pittsburgh, PA, May 15-20, 1999, ACM Press, 96-103.

43. Siewiorek, D., Smailagic, A., Bass, L., Siegel, J., Martin, R. and Bennington, B. Adtranz: A Mobile Computing System for Maintenance and Collaboration. *The Second International Symposium on Wearable Computers*, Pittsburgh, PA, October 1998, 25-40.

44. Suchman, L. *Plans and Situated Actions*. New York, NY: Cambridge Press, 1987.

45. Stein, R., Ferrero, S., Hetfield, M., Quinn, A., Krichever, M., Development of a Commercially Successful Wearable Data Collection System. *The Second International Symposium on Wearable Computers*, Pittsburgh, PA, October 1998, 18-24.

46. Streitz, N. A., Geißler, J., Holmer, T., Honomi, S., Müller-Tomfelde, C., Reische, W., Rexroth, P, Seitz, Pl, and Steinmetz, R. i-LAND: An interactive Landscape for Creativity and Innovation, *Proceedings of CHI'99*, Pittsburgh, PA, May 15-20, 1999, ACM Press, 120-127.

47. Weiser, M. The computer for the 21st Century. *Scientific American*, 3, September 1991. 94-104.

48. Weiser, M. Some Computer Science Issues in Ubiquitous Computing, *Communications of the ACM*, Vol. 36(7), 1993, 75-84.

49. Yan, J., Stiefelhagen, Meier, U., and Waibel, A. Visual Tracking for Multimodal Human Computer Interaction. Proceeding of CHI '98, Los Angeles, Ca., April 1998. 140-147.

50. Universal Plug and Play Forum web site: http://www.upnp.org/

51. T. Höllerer, S. Feiner, T. Terauchi, G. Rashid, D. Hallaway, *Exploring MARS: Developing Indoor and Outdoor User Interfaces to a Mobile Augmented Reality System*, In: Computers and Graphics, 23(6), Elsevier Publishers, Dec. 1999, pp. 779-785